bdiCW

BDM interface for CodeWarrior™ Debugger

PowerPC MPC8xx/MPC5xx

User Manual

Manual Version 1.01 for BDI1000

ABATRON

© 1992-2001 ABATRON AG

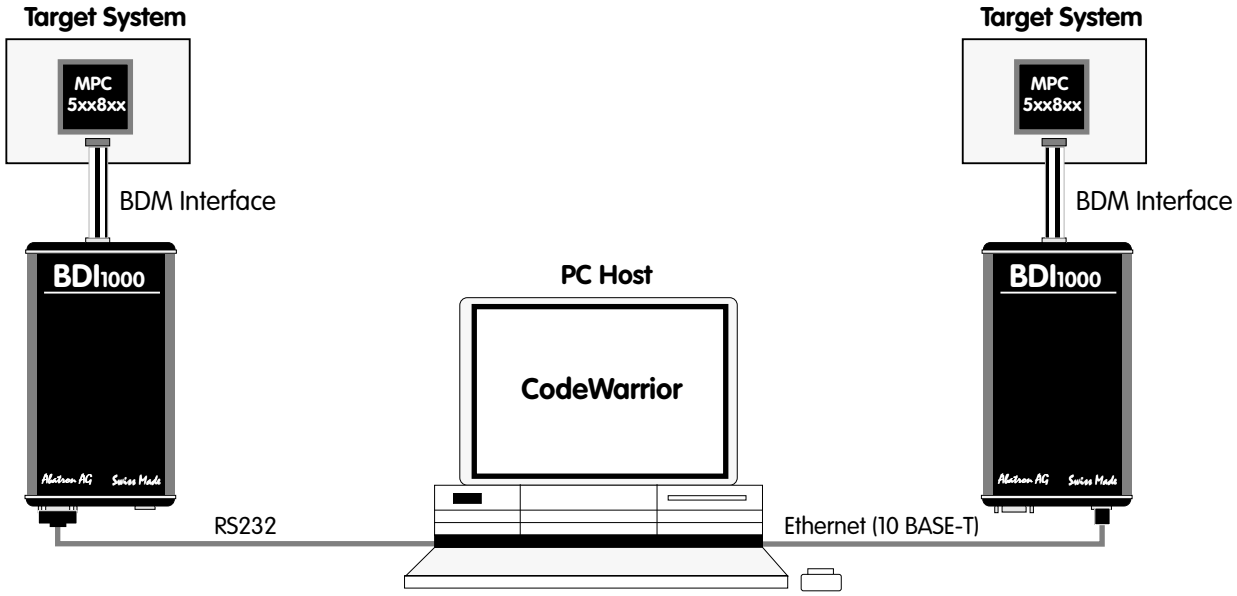
1 Introduction 3
1.1 BDI1000 3
2 Installation 4
2.1 Connecting the BDI1000 to the Target 4
2.1.1 Changing Target Processor Type 6
2.2 Connecting the BDI1000 to Power Supply 7
2.2.1 External Power Supply 7
2.2.2 Power Supply from Target System 8
2.3 Status LED «MODE» 9
2.4 Connecting the BDI1000 to Host 10
2.4.1 Serial line communication 10
2.4.2 Ethernet communication 11
2.5 Installation of the Configuration Software 12
2.6 Configuration 13
2.6.1 BDI1000 Setup/Update 13
3 Init List 15
4 BDI working modes 16
4.1 Startup Mode 17
4.1.1 Startup mode RESET 17
4.1.2 Startup Mode STOP 17
4.1.3 Startup mode RUN 17
5 Working with CodeWarrior 18
5.1 Setup 18
5.2 Exceptions 18
6 Specifications 19
7 Environmental notice 20
8 Declaration of Conformity (CE) 20
9 Warranty 21

Appendices

A Troubleshooting 22
B Maintenance 23
C Trademarks 25

© Copyright 1992-2001 by ABATRON AG

1 Introduction



The BDI1000 adds Background Debug Mode features to the CodeWarrior debugger environment from Metrowerks. With the BDI1000, you control and monitor the microcontroller solely through the stable on-chip debugging services. You won't waste time and target resources with a software ROM monitor, and you eliminate the cabling problems typical of ICE's. This combination runs even when the target system crashes and allows developers to continue investigating the cause of the crash.
A RS232 interface with a maximum of 115 kBaud and a 10Base-T Ethernet interface is available for the host interface.
The configuration software is used to update the firmware and to configure the BDI1000 so it works with the CodeWarrior debugger.

1.1 BDI1000

The BDI1000 is a processor system in a small box. It implements the interface between the BDM pins of the target CPU and a 10Base-T Ethernet / RS232 connector. BDI1000 is powered by a MC68331, 256Kbyte RAM and a flash memory of 512Kbyte. As a result of consistent implementation of lasted technology, the BDI1000 is optimally prepared for further enhancements. The firmware and the programmable logic of the BDI1000 can be updated by the user with a simple Windows based configuration program. The BDI1000 supports target system voltages from 2.7 up to 5 Volts.

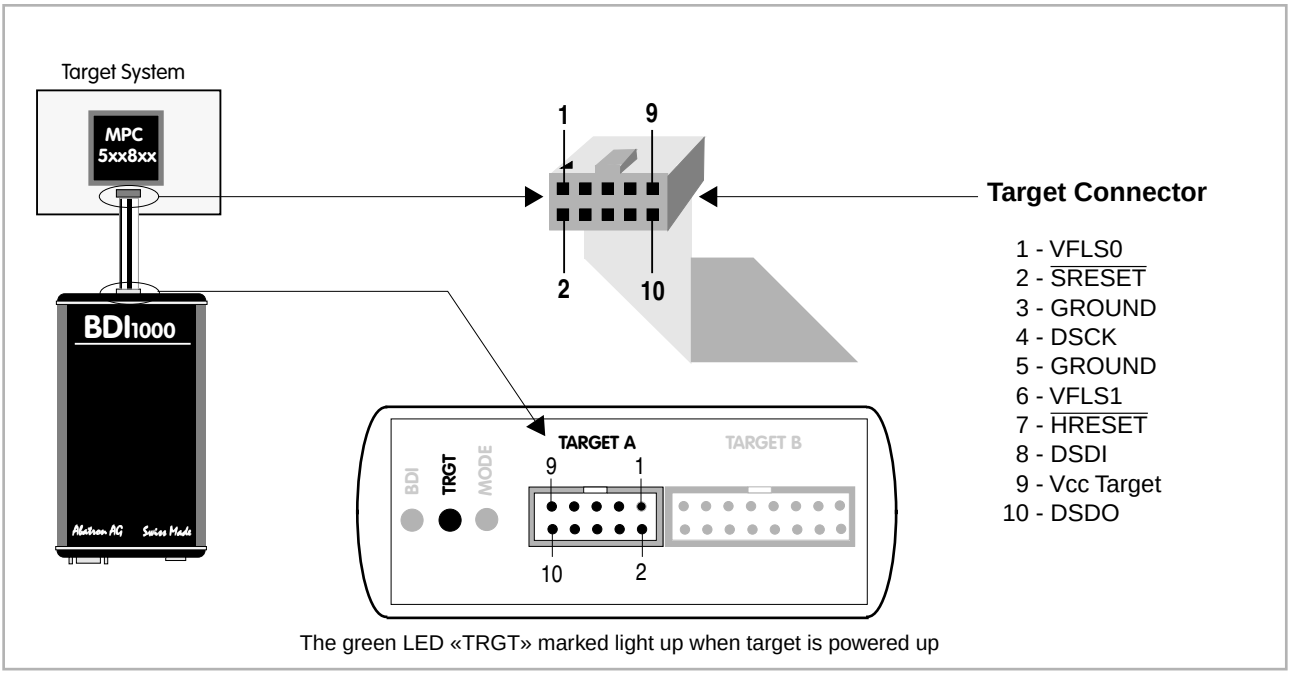
© Copyright 1992-2001 by ABATRON AG

2 Installation

2.1 Connecting the BDI1000 to the Target

The cable to the target system is a ten pin flat ribbon cable. In case where the target system has an appropriate connector, the cable can be directly connected. The pin assignment is in accordance with the Motorola specification.

In order to ensure reliable operation of the BDI (EMC, runtimes, etc.) the target cable length must not exceed 20 cm (8").



The green LED «TRGT» marked light up when target is powered up

For TARGET A connector signals see table on next page.

© Copyright 1992-2001 by ABATRON AG

BDI MAIN / TARGET A Connector Signals:

Pin	Name	Description
1	VFLS0	These pin and pin 6 (VFLS1) indicate to the debug port controller whether or not the MPC is in debug mode. When both VFLS0 and VFLS1 are at "1", the MPC is in debug mode.
2	$\overline{\text{SRESET}}$	This is the Soft-Reset bidirectional signal of the MPC8xx. On the MPC5xx it is an output. The debug port configuration is sampled and determined on the rising-edge of $\overline{\text{SRESET}}$ (for both processor families). On the MPC8xx it is a bidirectional signal which may be driven externally to generate soft reset sequence. This signal is in fact redundant regarding the MPC8xx debug port controller since there is a soft-reset signal integrated within the debug port protocol. However, the local debug port controller uses this signal for compatibility with MPC5xx existing boards and s/w.
3+5	GND	**System Ground**
4	DSCK	**Debug-port Serial Clock** During asynchronous clock mode, the serial data is clocked into the MPC according to the DSCK clock. The DSCK serves also a role during soft-reset configuration.
6	VFLS1	These pin and pin 1 (VFLS0) indicate to the debug port controller whether or not the MPC is in debug mode. When both VFLS0 and VFLS1 are at "1", the MPC is in debug mode.
7	$\overline{\text{HRESET}}$	This is the Hard-Reset bidirectional signal of the MPC. When this signal is asserted (low) the MPC enters hard reset sequence which include hard reset configuration. This signal is made redundant with the MPC8xx debug port controller since there is a hard-reset command integrated within the debug port protocol.
8	DSDI	**Debug-port Serial Data In** Via the DSDI signal, the debug port controller sends its data to the MPC. The DSDI serves also a role during soft-reset configuration.
9	Vcc Target	**TARGET POWER** This input to the BDI1000 is used to detect if the target is powered up.
10	DSDO	**Debug-port Serial Data Out** DSDO is clocked out by the MPC according to the debug port clock, in parallel with the DSDI being clocked in. The DSDO serves also as "READY" signal for the debug port controller to indicate that the debug port is ready to receive controller's command (or data).

Mention of sources used: MPC860ADS User's Manual, Revision A

Enhanced Debug Mode Detection:

For MPC8xx and MPC555 targets, debug mode (Freeze) detection also works when the BDM connector pins VFLS0 and VFLS1 are not connected to the target. If not connected to VFLSx, this BDM connector pins should be left open or tied to Vcc. The BDI uses the following algorithm to check if the target is in debug mode (freezed):

```
BOOL PPC_TargetFreezed(void) {
  if ((VFLS0 != 1) | (VFLS0 != 1)) return FALSE;
  read debug port status;
  if (status == freezed) return TRUE;
  else                   return FALSE;
```

© Copyright 1992-2001 by ABATRON AG

2.1.1 Changing Target Processor Type

Before you can use the BDI1000 with an other target processor type (e.g. CPU32 <--> PPC), a new setup has to be done (see chapter 2.6 «Configuration»). During this process the target cable must be disconnected from the target system. The BDI1000 needs to be supplied **between 2.5V and 5V** via the POWER connector. For more information see chapter 2.2.1 «External Power Supply».

To avoid data line conflicts, the BDI1000 must be disconnected from the target system while programming the logic for an other target CPU.

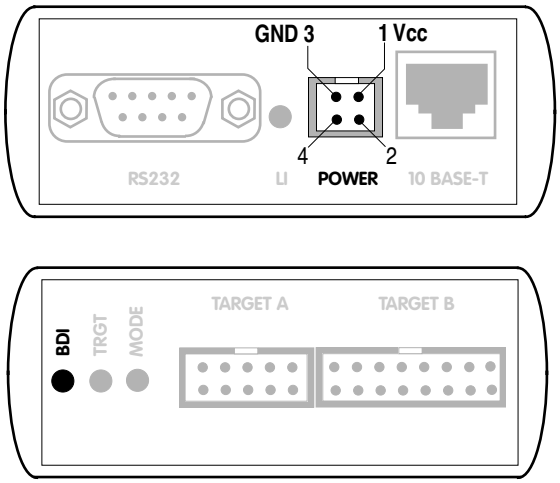
© Copyright 1992-2001 by ABATRON AG

2.2 Connecting the BDI1000 to Power Supply

2.2.1 External Power Supply

The BDI1000 needs to be supplied **between 2.5V and 5V** via the POWER connector. The available power supply from Abatron (option) or the enclosed power cable can be directly connected. In order to ensure reliable operation of the BDI1000, keep the power supply cable as short as possible.

For error-free operation, the power supply to the BDI1000 must be between 2.5V and 5V DC. **The maximal tolerable supply voltage is 5.25 VDC. Any higher voltage or a wrong polarity might destroy the electronics.**



POWER Connector

1 - Vcc (+2.5 ... +5V)
2 - VccTGT
3 - GROUND
4 - NOT USED

The green LED «BDI» marked light up when power (2.5 – 5V) is connected to the BDI1000

Please switch on the system in the following sequence:

- 1 --> external power supply
- 2 --> target system

© Copyright 1992-2001 by ABATRON AG

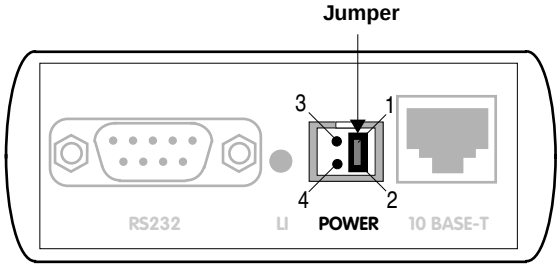
2.2.2 Power Supply from Target System

The BDI1000 needs to be supplied between 2.5V and 5V via TARGET A connector. This mode can only be used when the target system runs **between 2.5V and 5V** and the pin «Vcc Target» is able to deliver a current up to:

- 900mA@2.5Vcc Target
- 700mA@3.3Vcc Target
- 450mA@5.0Vcc Target

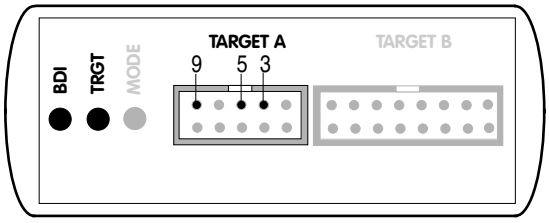
For pin description and layout see chapter 2.1 «Connecting the BDI1000 to Target». Insert the enclosed Jumper as shown in figure below. **Please ensure that the jumper is inserted correctly**.

For error-free operation, the power supply to the BDI1000 must be between 2.5V and 5V DC. **The maximal tolerable supply voltage is 5.25 VDC. Any higher voltage or a wrong polarity might destroy the electronics.**



POWER Connector

1 - Vcc BDI1000 (+2.5 ... +5V)
2 - Vcc Target (+2.5 ... +5V)
3 - GROUND
4 - NOT USED



The green LEDs «BDI» and «TRGT» marked light up when target is powered up and the jumper is inserted correctly

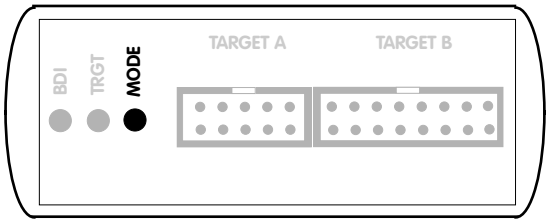
Target A Connector

1 - VFLS0
2 - $\overline{\text{SRESET}}$
3 - GROUND
4 - DSCK
5 - GROUND
6 - VFLS1
7 - $\overline{\text{HRESET}}$
8 - DSDI
9 - Vcc Target
10 - DSDO

© Copyright 1992-2001 by ABATRON AG

2.3 Status LED «MODE»

The built in LED indicates the following BDI states:



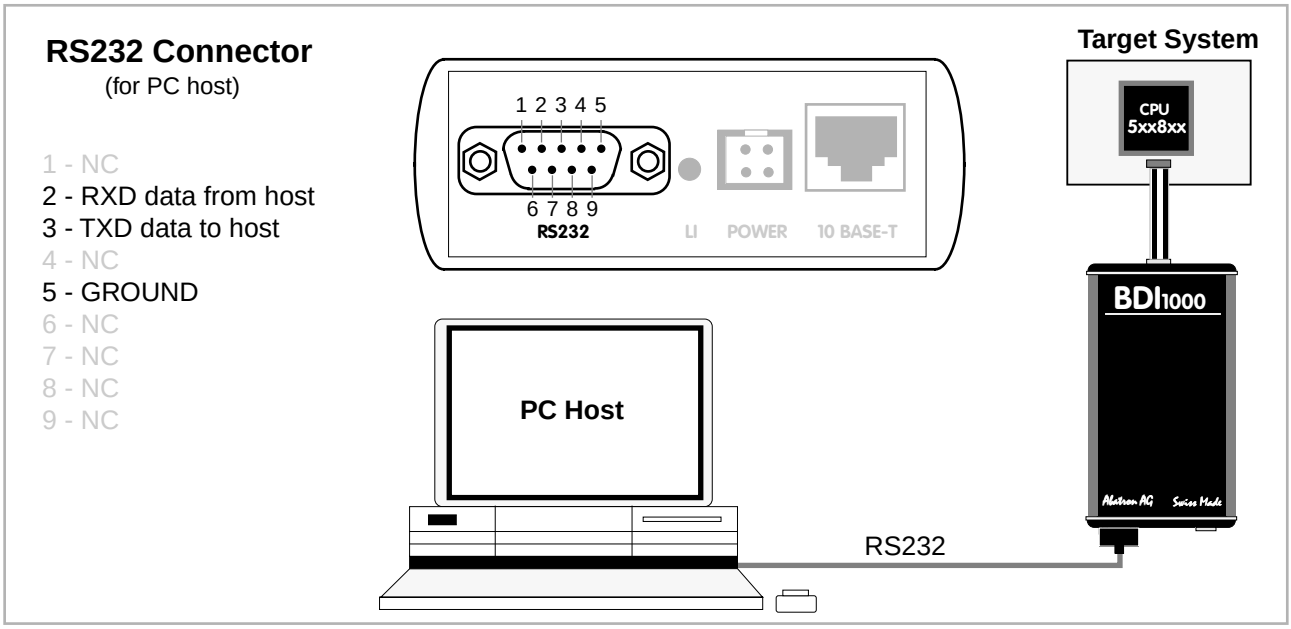
MODE LED	BDI STATES
OFF	The BDI is ready for use, the firmware is already loaded.
ON	The power supply for the BDI1000 is < 2.5VDC.
BLINK	The BDI «loader mode» is active (an invalid firmware is loaded or loading firmware is active).

© Copyright 1992-2001 by ABATRON AG

2.4 Connecting the BDI1000 to Host

2.4.1 Serial line communication

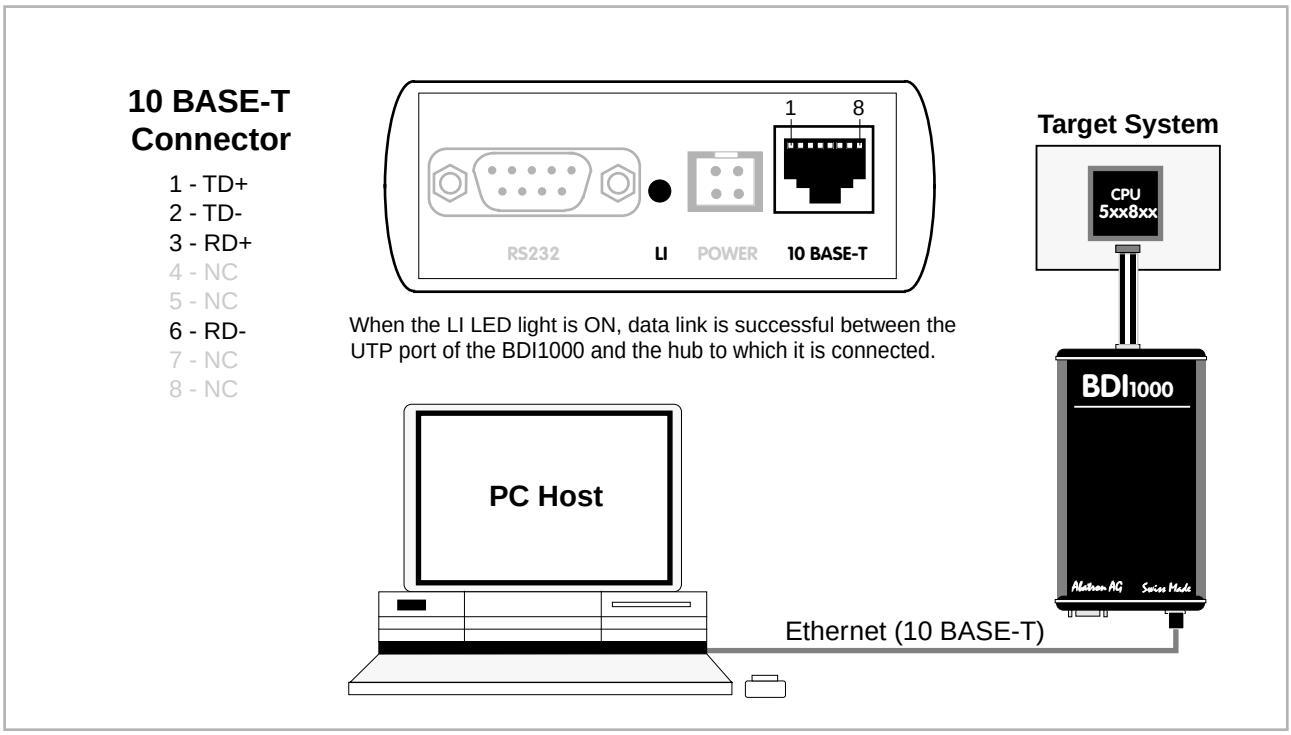
The host is connected to the BDI through the serial interface (COM1...COM4). The communication cable between BDI and Host is a serial cable (RXD / TXD are crossed). There is the same connector pinout for the BDI and for the Host side (Refer to Figure below).



© Copyright 1992-2001 by ABATRON AG

2.4.2 Ethernet communication

The BDI1000 has a built-in 10 BASE-T Ethernet interface (see figure below). Connect an UTP (Unshilded Twisted Pair) cable to the BD1000. For thin Ethernet coaxial networks you can connect a commercially available media converter (BNC --> 10 BASE-T) between your network and the BDI1000. Contact your network administrator if you have questions about the network.



© Copyright 1992-2001 by ABATRON AG

2.5 Installation of the Configuration Software

On the enclosed diskette you will find the BDI configuration software and the firmware required for the BDI. Copy all these files to a directory on your hard disk.

The following files are on the diskette:

b10mpc.exe	Configuration program
b10mpc.hlp	Helpfile for the configuration program
b10mpc.cnt	Help contents file
b10ppcfw.xxx	Firmware for BDI1000 for MPC8xx/MPC5xx targets
ppcjed10.xxx	JEDEC file for the BDI logic device programming
bdiifc32.dll	BDI Interface DLL
*.bdi	Configuration Examples

Example of an installation process:

- Copy the entire contents of the enclosed diskette into a directory on the hard disk.
- You may create a new shortcut to the b10mpc.exe configuration program.
- Copy the BDI interface DLL to ...\CodeWarrior\Bin\Plugins\Support\Abatron\bdiifc32.dll

© Copyright 1992-2001 by ABATRON AG

2.6 Configuration

Before you can use the BDI together with the debugger, the BDI must be configured. Use the *SETUP* menu and follow the steps listed below:

- Load or update the firmware / logic, store IP address --> *Firmware*
- Set the communication parameters between Host and BDI --> *Communication*
- Setup an initialization list for the target processor --> *Initlist*
- Select the working mode --> *Mode*
- Transmit the configuration to the BDI --> *Mode Transmit*

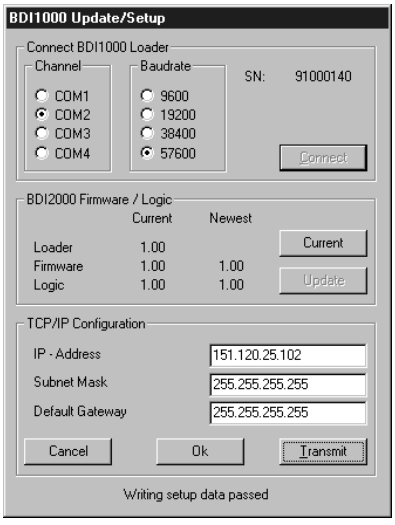
For information about the dialogs and menus use the help system (F1).

2.6.1 BDI1000 Setup/Update

First make sure that the BDI is properly connected (see Chapter 2.1 to 2.4). The BDI must be connected via RS232 to the Windows host.

To avoid data line conflicts, the BDI1000 must be disconnected from the target system while programming the logic for an other target CPU (see Chapter 2.1.1).

The following dialogbox is used to check or update the BDI firmware and logic and to set the network parameters.



dialog box «BDI1000 Update/Setup»

The following options allow you to check or update the BDI firmware and logic and to set the network parameters:

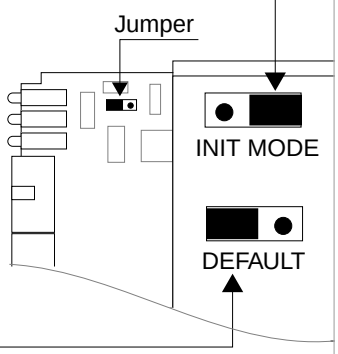
Channel	Select the communication port where the BDI1000 is connected during this setup session.
Baudrate	Select the baudrate used to communicate with the BDI1000 loader during this setup session.

© Copyright 1992-2001 by ABATRON AG

Connect	Click on this button to establish a connection with the BDI1000 loader. Once connected, the BDI1000 remains in loader mode until it is restarted or this dialog box is closed.
Current	Press this button to read back the current loaded BDI1000 software and logic versions. The current loader, firmware and logic version will be displayed.
Update	This button is only active if there is a newer firmware or logic version present in the execution directory of the BDI setup software. Press this button to write the new firmware and/or logic into the BDI1000 flash memory / programmable logic.
IP Address	Enter the IP address for the BDI1000. Use the following format: xxx.xxx.xxx.xxxe.g.151.120.25.101 Ask your network administrator for assigning an IP address to this BDI1000. Every BDI1000 in your network needs a different IP address.
Subnet Mask	Enter the subnet mask of the network where the BDI is connected to. Use the following format: xxx.xxx.xxx.xxxe.g.255.255.255.0 A subnet mask of 255.255.255.255 disables the gateway feature. Ask your network administrator for the correct subnet mask.
Default Gateway	Enter the IP address of the default gateway. Ask your network administrator for the correct gateway IP address. If the gateway feature is disabled, you may enter 255.255.255.255 or any other value..
Transmit	Click on this button to store the network configuration in the BDI1000 flash memory.

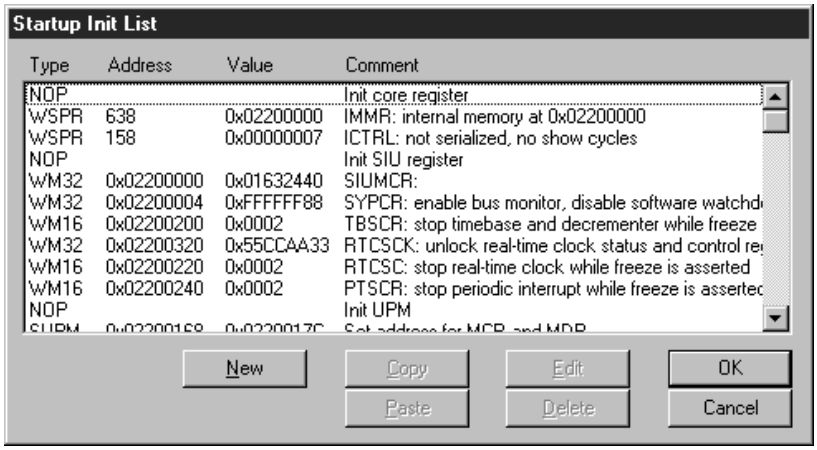
In rare instances you may not be able to load the firmware in spite of a correctly connected BDI (error of the previous firmware in the flash memory). **Before carrying out the following procedure, check the possibilities in Appendix «Troubleshooting»**. In case you do not have any success with the tips there, do the following:

- Switch OFF the power supply for the BDI and open the unit as described in Appendix «Maintenance»
- Place the jumper in the **«INIT MODE»** position
- Connect the power cable or target cable if the BDI is powered from target system
- Switch ON the power supply for the BDI again and wait until the LED «MODE» blinks fast
- Turn the power supply OFF again
- Return the jumper to the **«DEFAULT»** position
- Reassemble the unit as described in Appendix «Maintenance»



© Copyright 1992-2001 by ABATRON AG

3 Init List


Startup Init List
Type Address Value Comment
NOP Init core register
WSPR 638 0x02200000 IMMR: internal memory at 0x02200000
WSPR 158 0x00000007 ICTRL: not serialized, no show cycles
NOP Init SIU register
WM32 0x02200000 0x01632440 SIUMCR:
WM32 0x02200004 0xFFFFFF88 SYPCR: enable bus monitor, disable software watchd
WM16 0x02200200 0x0002 TBSCR: stop timebase and decrementer while freeze
WM32 0x02200320 0x55CCAA33 RTCSCK: unlock real-time clock status and control re
WM16 0x02200220 0x0002 RTCSC: stop real-time clock while freeze is asserted
WM16 0x02200240 0x0002 PTSCR: stop periodic interrupt while freeze is asserted
NOP Init UPM
New
Copy
Edit
OK
Paste
Delete
Cancel

dialog box «Startup Init List»

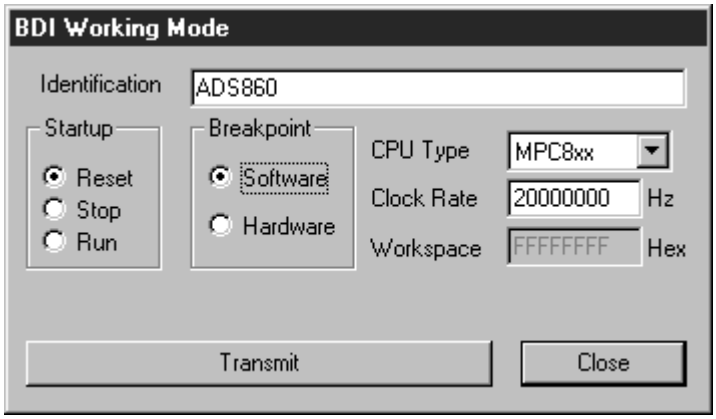
In order to prepare the target for debugging, you can define an Initialization List. This list is stored in the Flash memory of the BDI1000 and worked through every time the target comes out of reset. Use it to get the target operational after a reset. The memory system is usually initialized through this list. After processing the init list, the RAM used to download the application must be accessible.

Use on-line help (F1) and the supplied configuration examples on the distribution disk to get more information about the init list.

Note:
You may also use the debuggers feature to setup the hardware. But keep in mind, that the BDI will speed up BDM communication clock after processing its own initialization list based on the "Clock Rate" field in the "BDI Working Mode" dialog box (see next chapter). The "Clock Rate" value you have to enter is therefore the value the target runs immediately after reset.
To use a BDM speed as fast as possible, you should at least speed up the target with an entry in the BDI initialization list and set the "Clock Rate" field to the appropriate value. The rest of the initialization (e.g. memory controller) can be done with the debugger.

© Copyright 1992-2001 by ABATRON AG

4 BDI working modes



dialog box «BDI Working Mode»

With this dialog box you can define how the BDI interacts with the target system.

Identification	Enter a text to identify this setup. This text can be read by the debugger with the appropriate Command.
Startup	Startup mode defines how the BDI interacts with the target processor after reset or power up. The options RESET, STOP or RUN can be selected.
Breakpoint	Breakpoint mode defines how instruction breakpoints are implemented. When Software is selected (default), instruction breakpoints are set as requested by the debugger (Software or Hardware breakpoints). When Hardware is selected, the BDI uses always hardware breakpoints. This is useful when the attached debugger does not support hardware breakpoints on instruction access. For CodeWarrior, select always Software.
CPU Type	Select the CPU type of the target system.
Clock Rate	Enter the clock rate the target CPU runs after BDI has worked through the init list. BDI selects the BDM communication speed based on this parameter. If this parameter selects a CPU clock rate that is higher than the real clock rate, BDM communication may fail. When selecting a clock rate slower than possible, BDM communication still works but not as fast as possible.
Workspace	In order to access the floating-point registers of a MPC5xx microprocessor, the BDI needs a workspace of 8 bytes in target RAM. Enter the base address of this RAM area. This memory is used, when a floating-point register is accessed. If there is no RAM space available for the BDI, you may enter 0xFFFFFFFF as the workspace address but then, accessing floating-point registers is not possible.
Transmit	Click on this button to send the initialization list and the working mode to the BDI. This is normally the last step done before the BDI can be used with the debugging system.

© Copyright 1992-2001 by ABATRON AG

4.1 Startup Mode

Startup mode defines how the BDI interacts with the target system after a reset or power up sequence.

4.1.1 Startup mode RESET

In this mode no ROM is required on the target system. The necessary initialization is done by the BDI with the programmed init list. The following steps are executed by the BDI after system reset or system power up:

- HRESET is activated on the target system.
- HRESET is deactivated and the target is forced into debug mode.
- The BDI works through the initialization list and writes to the corresponding addresses.

The RESET mode is the standard working mode. Other modes are used in special cases (i.e. applications in ROM, special requirements on the reset sequence...).

4.1.2 Startup Mode STOP

In this mode the initialization code is in a ROM on the target system. The code in this ROM handles base initialization. At the end of the code, the initialization program enters an endless loop until it is interrupted by the BDI. This mode is intended for special requirements on the reset sequence (e.g. loading a RAM based programmable logic device).
In this mode the following steps are executed by the BDI after system reset or power up:

- HRESET is activated on the target system.
- HRESET is deactivated and the target is forced into debug mode.
- The target is started and begins executing application code.
- After a delay of 2 seconds, the target is forced into debug mode.
- The BDI works through the initialization list and writes the corresponding addresses.

4.1.3 Startup mode RUN

This mode is used to debug applications which are already stored in ROM. The application is started normally and is stopped when the debugger is started.
In this mode, the following steps are executed by the BDI after system reset or power up:

- HRESET is activated on the target system.
- HRESET is deactivated and the target is forced into debug mode.
- The target is startet and begins executing application code.
- The application runs until it is stopped by the debugger.

© Copyright 1992-2001 by ABATRON AG

5 Working with CodeWarrior

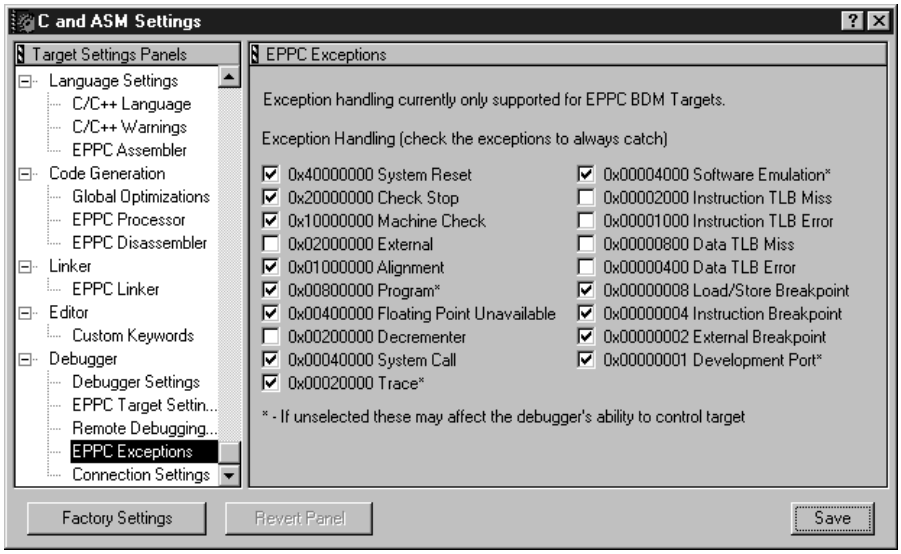
5.1 Setup

Use the CodeWarrior "IDE Preferences" dialog box, section "Debugger -> Remote Connections", and setup the appropriate communication parameters for the "Abatron TCP/IP" and "Abatron Serial" connection.

Use the CodeWarrior "your project Settings" dialog box, section "Debugger -> Remote Debugging" and select either "Abatron TCP/IP" or "Abatron Serial".

5.2 Exceptions

From the CodeWarrior menu "Edit" open "your project Settings...".
Select "Debugger - EPPC Exceptions". This opens the "EPPC Exceptions" dialog box.
Here it is possible to define which EPPC exceptions causes an entry into debug mode. Do not check exceptions which should be handled by application code (typically External Interrupt and Decrementer Interrupt). Never disable Development Port and External Breakpoint. For normal debugging you should also not disable Program, Trace, Software Emulation, Load/Store Breakpoint and Instruction Breakpoint.



© Copyright 1992-2001 by ABATRON AG

6 Specifications

Operating Voltage Limiting	2.5 ... 5.25VDC
Power Supply Current (max)	900mA@2.5V 700mA@3.3V 450mA@5.0V
RS232 Interface: Baud Rates	9’600,19’200, 38’400, 57’600,115’200
Data Bits	8
Parity Bits	none
Stop Bits	1
Network Interface	10 BASE-T
Serial Transfer Rate between BDI and Target	5.5 Mbit/s (BDM) 12 Mbit/s (JTAG)
Supported target voltage	2.7 ... 5 VDC
Operating Temperature	+ 5 °C ... +60 °C
Storage Temperature	-20 °C ... +65 °C
Relative Humidity (noncondensing)	<90 %rF
Size	160 x 85 x 35 mm
Weight (without cables)	280 g
Electromagnetic Compatibility (EMC)	EN 50081-2, EN 50082-2

Specifications subject to change without notice

© Copyright 1992-2001 by ABATRON AG

7 Environmental notice

Disposal of the equipment must be carried out at a designated disposal site.

8 Declaration of Conformity (CE)

CE

Declaration of Conformity

This declaration is valid for the following product:

Type of device: **BDM/JTAG Interface**
Product name: **BDI1000**

The signing authorities state, that the above mentioned equipment meets the requirements for emission and immunity according to

EMC Directive 89/336/EEC

The evaluation procedure of conformity was assured according to the following standards:

EN50081-2
EN50082-2

This declaration of conformity is based on the test report no. QNL-E079-05-9-a of Quinel, Zug, accredited according to EN 45001.

Manufacturer:

Abatron AG
Stöckenstrasse 4
CH-6221 Rickenbach

Authority:

Max Vock
Marketing Director

Ruedi Dummermuth
Technical Director

Rickenbach, November 2, 1999

© Copyright 1992-2001 by ABATRON AG

9 Warranty

ABATRON Switzerland warrants the physical diskette, cable, BDI1000 and physical documentation to be free of defects in materials and workmanship for a period of 24 months following the date of purchase when used under normal conditions.

In the event of notification within the warranty period of defects in material or workmanship, ABATRON will replace defective diskette, cable, BDI1000 or documentation. The remedy for breach of this warranty shall be limited to replacement and shall not encompass any other damages, including but not limited loss of profit, special, incidental, consequential, or other similar claims.
ABATRON Switzerland specifically disclaims all other warranties- expressed or implied, including but not limited to implied warranties of merchantability and fitness for particular purposes - with respect to defects in the diskette, cable, BDI1000 and documentation, and the program license granted herein, including without limitation the operation of the program with respect to any particular application, use, or purposes. In no event shall ABATRON be liable for any loss of profit or any other commercial damage, including but not limited to special, incidental, consequential, or other damages.

Failure in handling which leads to defects are not covered under this warranty. The warranty is void under any self-made repair operation except exchanging the fuse.

© Copyright 1992-2001 by ABATRON AG

Appendices

A Troubleshooting

Problem
The firmware can not be loaded.

Possible reasons

- The BDI is not correctly connected with the target system (see chapter 2).
- The power supply of the target system is switched off or not in operating range (2.5 VDC ... 5 VDC) --> MODE LED is OFF or RED
- The built in fuse is damaged --> MODE LED is OFF
- The BDI is not correctly connected with the Host (see chapter 2).
- A wrong communication port (Com 1...Com 4) is selected.

Problem
No working with the target system (loading firmware is ok).

Possible reasons

- Wrong pin assignment (BDM/JTAG connector) of the target system (see chapter 2).
- Target system initialization is not correctly --> enter an appropriate target initialization list.
- An incorrect IP address was entered (BDI1000 configuration)
- BDM/JTAG signals from the target system are not correctly (short-circuit, break, ...).
- The target system is damaged.

Problem
Network processes do not function (loading the firmware was successful)

Possible reasons

- The BDI1000 is not connected or not correctly connected to the network (LAN cable or media converter)
- An incorrect IP address was entered (BDI1000 configuration)

© Copyright 1992-2001 by ABATRON AG

B Maintenance

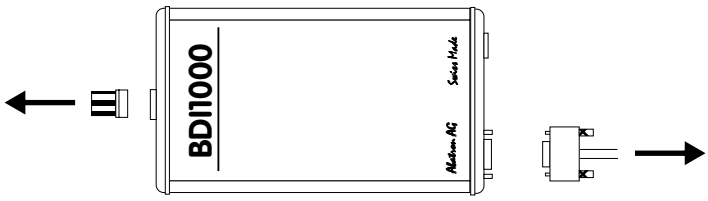
The BDI needs no special maintenance. Clean the housing with a mild detergent only. Solvents such as gasoline may damage it.

If the BDI is connected correctly and it is still not responding, then the built in fuse might be damaged (in cases where the device was used with wrong supply voltage or wrong polarity). To exchange the fuse or to perform special initialization, please proceed according to the following steps:

Observe precautions for handling (Electrostatic sensitive device)
Unplug the cables before opening the cover.
Use exact fuse replacement (Microfuse MSF 1.6 AF).

1

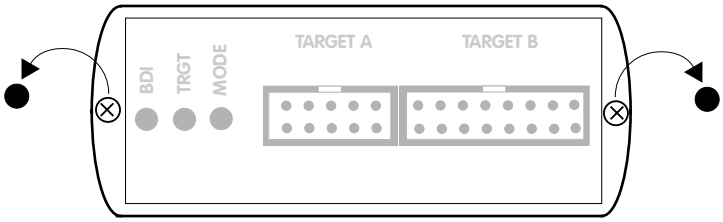
1.1 Unplug the cables



2

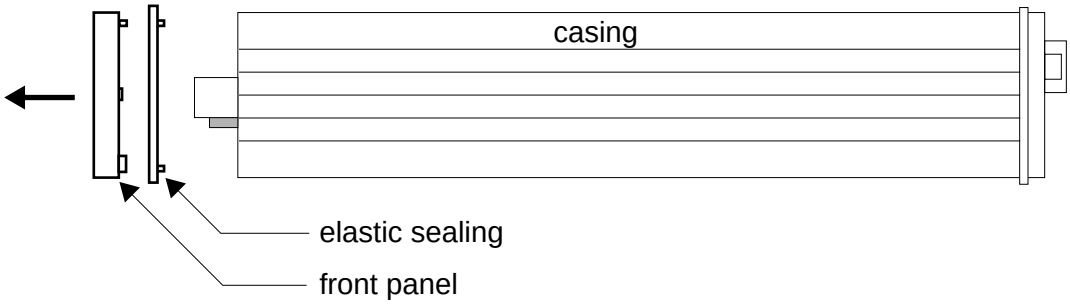
2.1 Remove the two plastic caps that cover the screws on target front side (e.g. with a small knife)

2.2 Remove the two screws that hold the front panel



3

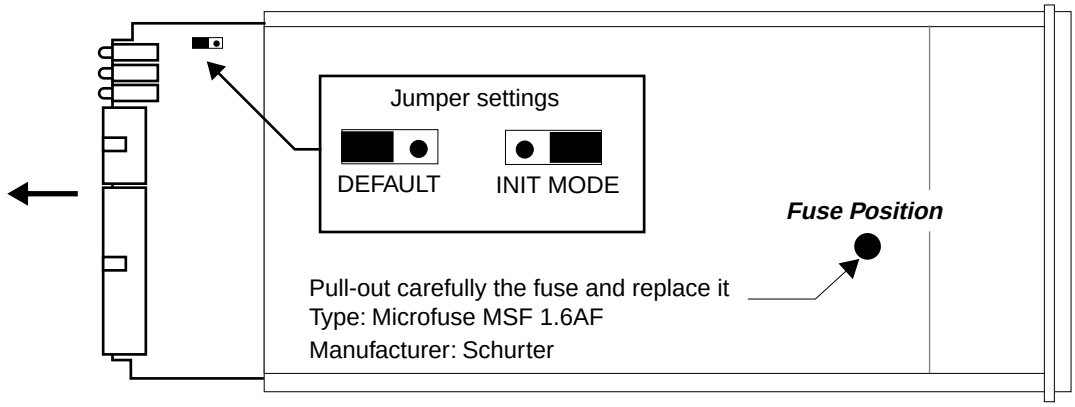
3.1 While holding the casing, remove the front panel and the blue elastig sealing



© Copyright 1992-2001 by ABATRON AG

4

4.1 While holding the casing, slide carefully the print in position as shown in figure below

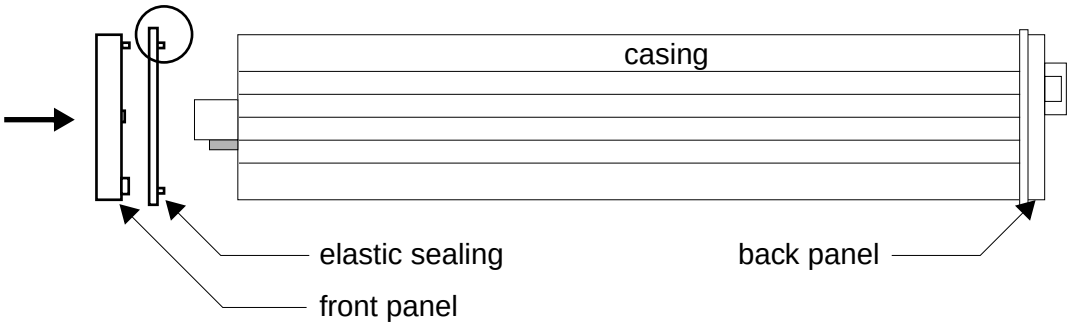


5

Reinstallation

5.1 Slide back carefully the print. Control that the LEDs align with the holes in the back panel.

5.2 Push carefully the front panel and the blue elastig sealing on the casing. Check that the LEDs align with the holes in the front panel and that the position of the sealing is as shown in the figure below.



5.3 Mount the screws (do not overtighten it)

5.4 Mount the two plastic caps that cover the screws

5.5 Plug the cables

Observe precautions for handling (Electrostatic sensitive device)
Unplug the cables before opening the cover.
Use exact fuse replacement (Microfuse MSF 1.6 AF).

© Copyright 1992-2001 by ABATRON AG

C Trademarks

All trademarks are property of their respective holders.

© Copyright 1992-2001 by ABATRON AG